Manual de Instalación y Configuración de un Servidor de dominio SAMBA-LDAP

Que es SAMBA

Samba es un producto que básicamente permite al sistema Unix conversar con sistemas Windows a través de la red de forma nativa. De esta forma, el sistema Unix aparece en el "Entorno de red", y clientes Windows pueden acceder a sus recursos de red e impresoras compartidas como si de otro sistema Windows se tratase.

Para ello, Samba implementa los protocolos NetBIOS y SMB.

NetBIOS es un protocolo de nivel de sesión que permite establecer sesiones entre dos ordenadores.

SMB (Server Message Block), implementado sobre NetBIOS, es el protocolo que permite a los sistemas Windows compartir archivos e impresoras .

El programa smbd se encarga de ofrecer los servicios de acceso remoto a ficheros e impresoras (implementando para ello el protocolo SMB), así como de autenticar y autorizar usuarios. smbd ofrece los dos modos de compartición de recursos existentes en Windows, basado en usuarios o basado en recursos.

En el modo basado en usuarios (propio de los dominios Windows NT o 2000) la autorización de acceso al recurso se realiza en función de nombres de usuarios registrados en un dominio, Mientras que en el modo basado en recursos (propio de Windows 3.11/95) a cada recurso se le asigna una contraseña, estando autorizado el acceso en función del conocimiento de dicha contraseña.

El programa nmbd permite que el sistema Unix participe en los mecanismos de resolución de nombres propios de Windows, lo cual incluye el anuncio en el grupo de trabajo, la gestión de la lista de computadores del grupo de trabajo, la contestación a peticiones de resolución de nombres y el anuncio de los recursos compartidos.

De esta forma, el sistema Unix aparece en el "Entorno de Red", como cualquier otro sistema Windows, publicando la lista de recursos que ofrece al resto de la red.

Adicionalmente a los dos programas anteriores, Samba ofrece varias utilidades. Algunas de las más relevantes son las siguientes:

- smbclient. Una interfaz similar a la utilidad ftp, que permite a un usuario de un sistema Unix conectarse a recursos SMB y listar, transferir y enviar archivos.
- swat (Samba Web Administration Tool). Esta utilidad permite configurar Samba de forma local o remota utilizando un navegador de web.
- smbfs Sistema de archivos SMB para Linux. Linux puede montar recursos SMB en su jerarquía, al igual que sucede con directorios compartidos vía NFS.
- winbind. Permite integrar un servidor Samba en un dominio Windows sin necesidad de crear usuarios Unix en el servidor Samba que correspondan con los usuarios del dominio Windows, simplificando así la labor de administración.

Que es SMB

Puesto que Samba es, fundamentalmente, una implementación para Unix del protocolo SMB, quizás la mejor forma de entender Samba es comenzar por describir SMB con un poco más de detalle.

SMB es un protocolo de comunicación de alto nivel que puede implementarse sobre diversos protocolos como TCP/IP, NetBEUI y IPX/SPX, junto con la ubicación de dichos protocolos en los niveles OSI y en la pila TCP/IP. Entre todas esas alternativas, tanto en el caso de Samba como de Windows 2000/XP, SMB se implementa habitualmente encima de NetBIOS sobre TCP/IP .

Niveles de Seguridad

Una de las consideraciones más importantes a la hora de configurar Samba es la selección del nivel de seguridad. Desde la perspectiva de un cliente, Samba ofrece dos modos de seguridad, denominados share y user:

Modo Share. En modo share, cada vez que un cliente quiere utilizar un recurso ofrecido por Samba, debe suministrar una contraseña de acceso asociada a dicho recurso. Una de las consideraciones más importantes a la hora de configurar Samba es la selección del nivel de seguridad.

Modo User. En modo user, el cliente debe establecer en primer lugar una sesión con el servidor Samba, para lo cual le suministra un nombre de usuario y una contraseña. Una vez Samba valida al usuario, el cliente obtiene permiso para acceder a los recursos ofrecidos por Samba.

En cualquiera de ambos, Samba tiene que asociar un usuario del sistema Unix en el que se ejecuta Samba con la conexión realizada por el cliente. Este usuario es el utilizado a la hora de comprobar los permisos de acceso a los archivos y directorios que el sistema Unix/Samba comparte en la red.

La selección del nivel de seguridad se realiza con la opción security, la cual pertenece a la sección [global]. Sus alternativas son las siguientes:

security = share | user | server | domain

Desde la perspectiva del cliente, el nivel share corresponde al modo de seguridad share y los niveles user, server y domain corresponden todos ellos al modo de seguridad user. A continuación se describen someramente los cuatro niveles.

El nivel share es utilizado normalmente en entornos en los cuales no existe un dominio Windows NT o 2000. En este caso, se asocia una contraseña por cada recurso, que debe proporcionarse correctamente desde el cliente cuando se pide la conexión.

En el nivel user, el encargado de validar al usuario es el sistema Unix donde Samba se ejecuta. La validación es idéntica a la que se realizaría si el usuario iniciase una sesión local en el ordenador Unix. Para que este método sea aplicable, es necesario que existan los mismos usuarios y con idénticas contraseñas en los sistemas Windows y en el sistema Unix donde Samba se ejecuta.

Desde la aparición de sistemas Windows como Windows 2000 y posteriores, la utilización de este nivel se ha vuelto complicada, ya que dichos sistemas Windows transmiten las contraseñas cifradas por la red. Puesto que Samba no posee acceso a las contraseñas cifradas por Windows, el sistema Unix ya no puede realizar la validación. Existen dos métodos para resolver este problema.

El primero consiste en modificar el registro del sistema Windows para permitir la transferencia de contraseñas sin cifrar por la red. El segundo método obliga a utilizar una tabla de contraseñas adicional en el sistema Unix, en la cual se almacenan las contraseñas cifradas de los usuarios Windows.

En el nivel server, Samba delega la validación del usuario en otro computador, normalmente un sistema Windows 2000. Cuando un cliente intenta iniciar una sesión con Samba, éste último intenta iniciar una sesión en el computador en el cual ha delegado la validación con la misma acreditación (usuario+contraseña) recibidos del cliente.

Si la sesión realizada por Samba es satisfactoria, entonces la solicitud del cliente es aceptada. Este método aporta la ventaja de no necesitar que las contraseñas se mantengan sincronizadas entre los sistemas Windows y Unix, ya que la contraseña Unix no es utilizada en el proceso de validación. Adicionalmente, no hay inconveniente en utilizar contraseñas cifradas, ya que la validación la realiza un sistema Windows 2000.

Por último, **el nivel domain**. Este nivel es similar al nivel server, aunque en este caso el computador en el que se delega la validación debe ser un DC, o una lista de Dcs. La ventaja de este método estriba en que el computador Samba pasa a ser un verdadero miembro del dominio W2000, lo que implica, por ejemplo, que puedan utilizarse las relaciones de confianza en las que participa el dominio W2000. Esto significa, en pocas palabras, que usuarios pertenecientes a otros dominios en los que los DCs confían son conocidos por Samba.

Que es LDAP

LDAP ("Lightweight Directory Acces Protocol", en español Protocolo Ligero de Acceso a Directorios) es un protocolo de tipo cliente-servidor para acceder a un servicio de directorio. Se usó inicialmente como un Front-end o interfaz final para x.500, pero también puede usarse con servidores de directorio únicos y con otros tipos de servidores de directorio.

Básicamente, OpenLDAP posee tres componentes principales:

- slapd - Dominio de servidor y herramientas
- Librerías que implementan el protocolo LDAP
- Programas cliente: ldapsearch, ldapadd, ldapdelete, entre otros

¿Qué es un directorio?

Un directorio es una base de datos, pero en general contiene información más descriptiva y más basada en atributos. La información contenida en un directorio normalmente se lee mucho más de lo que se escribe. Como consecuencia los directorios no implementan normalmente los complicados esquemas para transacciones o esquemas de reducción que las bases de datos utilizan para llevar a cabo actualizaciones complejas de grandes volúmenes de datos, Las actualizaciones en un directorio son

usualmente cambios sencillos de todo o nada, si es que permiten algo.

Un directorio LDAP es una base de datos?

El sistema gestor de una base de datos (DBMS) de Oracle ó Microsoft es usado para procesar peticiones (queries) ó actualizaciones a una base de datos relacional. Estas bases de datos pueden recibir miles de inserción, modificación o borrado por segundo. Un servidor LDAP es usado para procesar peticiones (queries) a un directorio LDAP. Pero LDAP procesa las órdenes de borrado y actualización de un modo muy lento. En otras palabras, LDAP es un tipo de base de datos, pero no es una base de datos relacional. No está diseñada para procesar miles de cambios por minuto como los sistemas relacionales, sino para realizar lecturas de datos de forma muy eficiente.

Funcionamiento de LDAP

- El servicio de directorio LDAP se basa en un modelo cliente-servidor.
- Uno o más servidores LDAP contienen los datos que conforman el árbol de directorio LDAP o base de datos troncal, el cliente LDAP se conecta con el servidor LDAP y le hace una consulta. El servidor contesta con la respuesta correspondiente, o bien con una indicación de donde puede el cliente hallar más información. No importa con que servidor LDAP se conecte el cliente ya que siempre observará la misma vista del directorio; el nombre que se le presenta a un servidor LDAP hace referencia a la misma entrada a la que haría referencia en otro servidor LDAP.
- Directorios de información. Por ejemplo bases de datos de empleados organizados por departamentos (siguiendo la estructura organizativa de la empresa) ó cualquier tipo de páginas amarilla.
- Sistemas de autenticación/autorización centralizada.
    - Active Directory Server, para gestionar todas las cuentas de acceso a una red corporativa y mantener centralizada la gestión del acceso a los recursos.
    - Sistemas de autenticación para páginas Web, algunos de los gestores de contenidos más conocidos disponen de sistemas de autenticación a través de LDAP.
    - Sistemas de control de entradas a edificios, oficinas .
- Sistemas de correo electrónico. Grandes sistemas formados por más de un servidor que accedan a un repositorio de datos común.
- Sistemas de alojamiento de páginas web y FTP, con el repositorio de datos de usuario compartido.
- Sistemas de autenticación basados en RADIUS, para el control de accesos de los usuarios a una red de conexión o ISP.
- Servidores de certificados públicos y llaves de seguridad.
- Autenticación única ó “single sign-on” para la personalización de aplicaciones.
- Perfiles de usuarios centralizados, para permitir itinerancia ó “Roaming”
- Libretas de direcciones compartidas.

Ejemplos de uso de LDAP

- **Sistema de correo electrónico**

  Cada usuario se identifica por su dirección de correo electrónico, los atributos que se guardan de cada usuario son su contraseña, su límite de almacenamiento (quota), la ruta del disco duro donde se almacenan los mensajes (buzón) y posiblemente atributos adicionales para activar sistemas anti-spam o anti-virus.

  Como se puede ver este sistema LDAP recibirá cientos de consultas cada día (una por cada email recibido y una cada vez que el usuario se conecta mediante POP3 o webmail). No obstante el número de modificaciones diarias es muy bajo, ya que solo se puede cambiar la contraseña o dar de baja al usuario, operaciones ambas que no se realizan de forma frecuente.

- **Sistema de autenticación a una red**

  Cada usuario se identifica por un nombre de usuario y los atributos asignados son la contraseña, los permisos de acceso, los grupos de trabajo a los que pertenece, la fecha de caducidad de la contraseña...

  Este sistema recibirá una consulta cada vez que el usuario acceda a la red y una más cada vez que acceda a los recursos del grupo de trabajo (directorios compartidos, impresoras...) para comprobar los permisos del usuario.

  Frente a estos cientos de consultas solo unas pocas veces se cambia la contraseña de un usuario o se le incluye en un nuevo grupo de trabajo.

Administración de Sistemas LDAP

Arquitectura de Funcionamiento

Backends, Vision General

Históricamente la arquitectura del servidor OpenLDAP (slapd, Standalone LDAP Daemon) fue dividida entre una sección frontal que maneja las conexiones de redes y el procesamiento del protocolo, y un base de datos dorsal o de segundo plano (backend) que trata únicamente con el almacenamiento de datos. La arquitectura es modular y una variedad de backends está disponible para interactuar con otras tecnologías, no sólo bases de datos tradicionales.

Actualmente 16 diferentes backends son proporcionados en la distribución de OpenLDAP, y varios proporcionados por terceros son conocidos para mantener otros backends de manera independiente. Los backends estándar están organizados de manera imprecisa en tres categorías:

- Backends de almacenamiento de datos (Data Storage backends) - estos realmente almacenan información .

- Proxy backends - actúan como puertas de enlace a otros sistemas de almacenamiento de datos .
- Backends dinámicos - estos generan datos sobre la marcha .

Backends de almacenamiento de datos (Data Storage backends) - estos realmente almacenan información

- back-bdb: el primer backend transaccional para OpenLDAP, construido en base a BerkeleyDB
- back-hdb: una variante de back-bdb que es totalmente jerárquica y soporta renombrado de sub-árboles
- back-ldif: construido en archivos LDIF de texto plano
- back-ndb: un backend transaccional construido en base al motor de cluster NDB de MySQL

Backends dinámicos - estos generan datos sobre la marcha

- back-config: configuración del servidor slapd vía LDAP
- back-dnssrv: localiza servidores LDAP vía DNS
- back-monitor: estadísticas de slapd vía LDAP
- back-null: un backend nulo, análogo a /dev/null en Unix
- back-perl: invoca arbitrariamente módulos de perl en respuesta a peticiones LDAP
- back-shell: invoca scripts de shell para peticiones LDAP
- back-sock: redirige peticiones LDAP sobre IPC a demonios de manera arbitraria

Arquitectura de Overlays

Generalmente una petición LDAP es recibida por el frontend, decodificada y luego transferida a un backend para procesamiento. Cuando el backend completa la petición, devuelve un resultado al frontend, quien luego envía el resultado al cliente LDAP. Un overlay es una pieza de código que puede ser insertada entre el frontend y el backend.

Es entonces capaz de interceptar peticiones y lanzar otras acciones en ellas antes de que el backend las reciba, y puede también actuar sobre los resultados del backend antes de que éstos alcancen el frontend. Overlays tiene acceso completo a las interfaces de programación (APIs) internas del servidor slapd, y por tanto pueden invocar cualquier llamada que podrían realizar el frontend u otros backends. Múltiples overlays pueden ser usados a la vez, formando una pila de módulos entre el frontend y el backend.

Proxy backends - actúan como puertas de enlace a otros sistemas de almacenamiento de datos

- back-ldap: proxy simple a otros servidores LDAP
- back-meta: proxy con características de meta-directorio
- back-passwd: usa un sistema basado en Unix de datos passwd y group
- back-relay: internamente redirige a otros backends de servidores slapd
- back-sql: establece conexiones a bases de datos SQL

Introducción a la estructura de árbol

Tradicionalmente se han usado las estructuras de árbol para jerarquizar la información contenida en un medio. El ejemplo más claro es la estructura de carpetas (directorios) de un sistema operativo. Esta organización nos permite ordenar la información en subdirectorios que contienen información muy específica.

Introducción a la estructura de árbol – Entradas

El modelo de información de LDAP está basado en entradas. Una entrada es una colección de atributos que tienen un único y global .

Nombre Distintivo (DN). El DN se utiliza para referirse a una entrada sin ambigüedades. Cada atributo de una entrada posee un tipo y uno o más valores. Los tipos son normalmente palabras nemotécnicas, como “cn” para common name, o “mail” para una dirección de correo.

La sintaxis de los atributos depende del tipo de atributo. Por ejemplo, un atributo cn puede contener el valor “Jose Manuel Suarez”. Un atributo email puede contener un valor “jmsuarez@ejemplo.com”. El atributo jpegPhoto ha de contener una fotografía en formato JPEG.

Introducción a la estructura de árbol – Atributos

Los datos del directorio se representan mediante pares de atributo y su valor. Por ejemplo el atributo commonName, o cn (nombre comun), se usa para almacenar el nombre de una persona. Puede representarse en el directorio a una persona llamada José Suarez mediante:

• cn: José Suarez

Cada persona que se introduzca en el directorio se define mediante la colección de atributos que hay en la clase de objetos person. Otros atributos:

• givenname: José □- --Atributo Requerido
• surname: Suarez
• mail: jmsuarez@ejemplo.com

Los atributos requeridos son aquellos que deben estar presentes en las entradas que utilicen en la clase de objetos. Todas las entradas precisas de los atributos permitidos son aquellos que pueden estar presentes en las entradas que utilicen la clase de objetos. Por ejemplo, en la clase de objetos person, Se requieren los atributos cn y sn.

Los atributos description (descripción), telephoneNumber (número de teléfono), seealso (véase también), y userpassword (contraseña del usuario) se permiten pero no son obligatorios.

Introducción a la estructura de árbol – Tipos de Atributos

Una definición de tipo de atributo especifica la sintaxis de un atributo y cómo se ordenan y comparan los atributos de ese tipo.

Los tipos de atributos en el directorio forman un árbol de clases. Por ejemplo, el tipo de atributo "commonName" es una subclase del tipo de atributo "name".

La estructura de árbol - El Archivo LDIF

Para importar y exportar información de directorio entre servidores de directorios basados en LDAP, o para describir una serie de cambios que han de aplicarse al directorio, se usa en general el archivo de formato conocido como LDIF (formato de intercambio de LDAP).

Un archivo LDIF almacena información en jerarquías de entradas orientadas a objeto. Todos los servidores LDAP que incluyen una utilidad para convertir archivos LDIF a formato orientadas a objeto. Normalmente es un archivo ASCII.

La estructura de árbol - Clases de Objetos

En LDAP, una clase de objetos define la colección de atributos que pueden usarse para definir una entrada. El estándar LDAP proporciona estos tipos básicos para las clases de objetos:

1. Grupos en el directorio, entre ellos listas no ordenadas de objetos individuales o de grupos de objetos.
2. Emplazamientos, como por ejemplo el nombre del país y su descripción.
3. Organizaciones que están en el directorio.
4. Personas que están en el directorio.

Una entrada determinada puede pertenecer a más de una clase de objetos. Por ejemplo, la entrada para personas se define mediante la clase de objetos person, pero también puede definirse mediante atributos en las clases de objetos inetOrgPerson, groupOfNames y organization. La estructura de clases de objetos del servidor determina la lista total de atributos requeridos y permitidos para una entrada concreta.

Pasos para la instalación y configuración de un Controlador Primario de Dominio (PDC) en Debian 5.0 mediante Samba, PAM/NSS y OpenLDAP

Introducción

Se explica en este articulo la implementación de una base de datos de usuarios centralizada mediante OpenLDAP, la implementación de un Controlador Primario de Dominio (PDC) mediante Samba, el cual utilizará la información del directorio LDAP para autenticar y asignar privilegios y restricciones, y la autenticación de usuarios desde terminales Linux/Unix mediante PAM/NSS .

Antes de comenzar se debe saber que PAM y NSS son los dos servicios que ofrecen mecanismos para identificar los usuarios no solo por el usuario local y el grupo de base de datos, sino también por administración de sistemas externos de usuario (por ejemplo, LDAP, MySQL ...). Ambos utilizan backends como sus fuentes de datos, y estos suelen tener los archivos de configuración separados.

Software a utilizar

* Debian GNU/Linux: en su versión 5.0.3.
* Samba: paquete samba, smbclient, smbfs y samba-doc en su versión 2:3.2.5-4lenny7
* OpenLDAP: paquete slapd en su versión 2.4.11-1+lenny1
* Ldap-utils: paquete ldap-utils en su versión 2.4.11-1+lenny1
* Libpam-ldap: paquete libpam-ldap en su versión 184-4.2
* Libnss-ldap: paquete libnss-ldap en su versión 261-2.1
* Smbldap-tools: paquete smbldap-tools en su versión 0.9.4-1

Se instalará todo el software de una vez, los paquetes de configuración de los diferentes servicios solicitarán cierta información, es recomendable responder con cualquier dato ya que mas adelante se procede a reconfigurar los paquetes y la información será solicitada de nuevo. Se puede, responder con datos reales si se cree conocer cuales son pero en este punto no es importante.

Instalar los paquetes necesarios:

aptitude install slapd ldap-utils libpam-ldap libpam-cracklib libnss-ldap samba samba-doc smbclient smbfs smbldap-tools libpam-dotfile

Configuración de SLAPD

Lo primero será reconfigurar el paquete slapd

dpkg-reconfigure slapd

Responder a las siguientes preguntas:

* ¿Omitir la configuración del servidor OpenLDAP?: No
* DNS Domain Name: dominio
* Organization Name: nombre de la organizacion
* Contraseña de Administrador: contraseña
* Verificar Contraseña: otra vez la contraseña
* Database backend to use: HDB
* ¿Desea que se borre la base de datos cuando se puergue el paquete slapd?: No

* ¿Desea mover la base de datos antigua?: Si
* Allow LDAPv2 protocol?: No

Para que ldap sea el soporte para samba se debe incorporar la estructura de grupos y usuarios que samba necesita, esa estructura debe contar con ciertos atributos y esos atributos se definen en un schema, pero antes de incorporar esos atributos se debe hacer una copia de la base de datos.

La herramienta slapcat permite volcar el contenido de una base de datos ldap a un archivo de texto ldif (LDAP Directory Interchange Format). Se hace un backup de la base de datos:

slapcat > ~/slapd.ldif

Agregar el Schema Samba al Directorio

Un schema (esquema) define el tipo de objetos que podemos manejar en nuestro arbol de directorio, sus atributos y sus reglas de sintaxis.

Slapd incluye por defecto los esquemas necesarios para almacenar informacion de cuentas Unix/Posix pero no incorpora soporte para el esquema de samba, afortunadamente el paquete samba-doc nos proveerá de uno. Simplemente haremos:

zcat /usr/share/doc/samba-doc/examples/LDAP/samba.schema.gz
> /etc/ldap/schema/samba.schema

El archivo /etc/ldap/slapd.conf

Es el archivo principal de configuración del servidor slapd. Se hace una copia de respaldo del mismo antes de modificar el original:

cp /etc/ldap/slapd.conf{,.original}

Para generar el password se utiliza la herramienta slappasswd, la clave que se utiliza es la misma que la que se ingresó cuando se configuró slapd: contraseña

No es requerimiento que sea la misma clave, pero se utiliza la misma para evitar confusiones

:~# slappasswd -h {md5}
New password: contraseña
Re-enter new password: otra vez la contraseña
{MD5}TmZgZ01/Z0/29bOPByMr4A==

La salida de slappasswd se utiliza como valor del parámetro rootpw en el archivo /etc/ldap/slapd.conf. Este parametro se establece para hacer posible la replicacion mediante syncrepl

Ahora reemplazar el contenido del archivo /etc/ldap/slapd.conf con lo siguiente (no olvidar reemplazar el valor del parámetro rootpw con su hash):

```
# This is the main slapd configuration file. See slapd.conf(5) for
more
# info on the configuration options.

######################################################
#################
# Global Directives:

# Features to permit
#allow bind_v2

# Schema and objectClass definitions
include         /etc/ldap/schema/core.schema
include         /etc/ldap/schema/cosine.schema
include         /etc/ldap/schema/nis.schema
include         /etc/ldap/schema/inetorgperson.schema
include         /etc/ldap/schema/samba.schema

# Where the pid file is put. The init.d script
# will not stop the server if you change this.
pidfile         /var/run/slapd/slapd.pid

# List of arguments that were passed to the server
argsfile        /var/run/slapd/slapd.args

# Read slapd.conf(5) for possible values
loglevel        none

# Where the dynamically loaded modules are stored
modulepath /usr/lib/ldap
moduleload back_hdb

# The maximum number of entries that is returned for a search
operation
sizelimit 500

# The tool-threads parameter sets the actual amount of cpu's that is
used
# for indexing.
tool-threads 1

######################################################
#################
# Specific Backend Directives for hdb:
# Backend specific directives apply to this backend until another
# 'backend' directive occurs
backend hdb

######################################################
```

```
#################
# Specific Backend Directives for 'other':
# Backend specific directives apply to this backend until another
# 'backend' directive occurs
#backend <other>

#######################################################
#################
# Specific Directives for database #1, of type hdb:
# Database specific directives apply to this databasse until another
# 'database' directive occurs
database        hdb

# The base of your directory in database #1
suffix          "dc=esdebian,dc=org"

# rootdn directive for specifying a superuser on the database. This
is needed
# for syncrepl.
rootdn          "cn=admin,dc=esdebian,dc=org"
rootpw          {MD5}TmZgZ01/Z0/29bOPByMr4A==

# Where the database file are physically stored for database #1
directory       "/var/lib/ldap"

# The dbconfig settings are used to generate a DB_CONFIG file the
first
# time slapd starts.  They do NOT override existing an existing
DB_CONFIG
# file.  You should therefore change these settings in DB_CONFIG
directly
# or remove DB_CONFIG and restart slapd for changes to take
effect.

# For the Debian package we use 2MB as default but be sure to
update this
# value if you have plenty of RAM
dbconfig set_cachesize 0 2097152 0

# Sven Hartge reported that he had to set this value incredibly high
# to get slapd running at all. See http://bugs.debian.org/303057 for
more
# information.

# Number of objects that can be locked at the same time.
dbconfig set_lk_max_objects 1500
# Number of locks (both requested and granted)
dbconfig set_lk_max_locks 1500
# Number of lockers
```

```
dbconfig set_lk_max_lockers 1500

# Indices to maintain for this database
index objectClass                       eq,pres
index ou,cn,sn,mail,givenname           eq,pres,sub
index uidNumber,gidNumber,memberUid     eq,pres
index loginShell                        eq,pres
## required to support pdb_getsampwnam
index uid                               pres,sub,eq
## required to support pdb_getsambapwrid()
index displayName                       pres,sub,eq
index nisMapName,nisMapEntry            eq,pres,sub
index sambaSID                          eq
index sambaPrimaryGroupSID              eq
index sambaDomainName                   eq
index default                           sub
index uniqueMember                      eq
index sambaGroupType                    eq
index sambaSIDList                      eq

# Save the time that the entry gets modified, for database #1
lastmod         on

# Checkpoint the BerkeleyDB database periodically in case of
system
# failure and to speed slapd shutdown.
checkpoint      512 30

# Where to store the replica logs for database #1
# replogfile /var/lib/ldap/replog

# users can authenticate and change their password
access                                                        to
attrs=userPassword,sambaNTPassword,sambaLMPassword,samba
PwdMustChange,sambaPwdLastSet
        by self write
        by anonymous auth
        by * none

# those 2 parameters must be world readable for password aging to
work correctly
# (or use a priviledge account in /etc/ldap.conf to bind to the
directory)
access to attrs=shadowLastChange,shadowMax
        by self write
        by * read

# all others attributes are readable to everybody
access to *
```

```
        by * read

# For Netscape Roaming support, each user gets a roaming
# profile for which they have write access to
#access to dn=".*,ou=Roaming,o=morsnet"
#       by dn="cn=admin,dc=example,dc=com" write
#       by dnattr=owner write

#######################################################################
# Specific Directives for database #2, of type 'other' (can be hdb too):
# Database specific directives apply to this databasse until another
# 'database' directive occurs
#database        <other>

# The base of your directory for database #2
#suffix "dc=debian,dc=org"
```

Como el archivo es muy extenso solo se analizara las diferencias entre el archivo original y el nuevo, se utilizara para eso diff, si no se conoce como funciona esta herramienta, bastará con saber que una linea que comienza con un + significa que ha sido agregada en el archivo nuevo y una que comienza con un - significa que la linea ha sido quitada, una linea sin nada significa que la linea permanece en ambos archivos.

Por partes:

```
include        /etc/ldap/schema/cosine.schema
include        /etc/ldap/schema/nis.schema
include        /etc/ldap/schema/inetorgperson.schema
+include        /etc/ldap/schema/samba.schema
```

Se ha agregado el esquema samba.schema para que sea incluido

```
-suffix          "dc=example,dc=com"
+suffix          "dc=esdebian,dc=org"
```

Se ha modificado el sufijo de la base de datos a esdebian.org

```
+rootdn          "cn=admin,dc=esdebian,dc=org"
+rootpw          {MD5}SCBBhdOlfHozkVbuKGZt5w==
```

Se han agregado las lineas que establecen el nombre (admin) y el password del administrador de la

base de datos.

```
-index          objectClass eq
+index objectClass                    eq,pres
+index ou,cn,sn,mail,givenname          eq,pres,sub
+index uidNumber,gidNumber,memberUid     eq,pres
+index loginShell                   eq,pres
+index uid                          pres,sub,eq
+index displayName                       pres,sub,eq
+index nisMapName,nisMapEntry           eq,pres,sub
+index sambaSID                     eq
+index sambaPrimaryGroupSID             eq
+index sambaDomainName                  eq
+index default                      sub
+index uniqueMember                     eq
+index sambaGroupType                   eq
+index sambaSIDList                     eq
```

Se ha agregado una gran cantidad de definiciones de índices. Al ser una base de datos, el incorporar índices acelerará las búsquedas lo cual es muy importante en un servidor atareado.

```
-access to attrs=userPassword,shadowLastChange
-       by dn="cn=admin,dc=example,dc=com" write
-       by anonymous auth
+access                                                          to
attrs=userPassword,sambaNTPassword,sambaLMPassword,samb
aPwdMustChange,sambaPwdLastSet
        by self write
+       by anonymous auth
        by * none
-access to dn.base="" by * read
+access to attrs=shadowLastChange,shadowMax
+       by self write
+       by * read
access to *
-       by dn="cn=admin,dc=example,dc=com" write
        by * read
```

Se otorgan permisos a los usuarios para acceder y modificar sus passwords, entre otras cosas.

Para verificar la correctitud del archivo de configuración se ejecuta la herramienta slaptest:

```
# slaptest -v -u
```

```
config file testing succeeded
```

Y ya podemos reiniciar el servido slapd

```
# /etc/init.d/slapd restart
```

Pruebas Preliminares de servidor ldap

Hagamos una consulta al servidor LDAP para ver si responde correctamente. Con el comando ldapsearch consultaremos el namingContexts

```
:~ # ldapsearch -x -b '' -s base '(objectclass=*)' namingContexts

# extended LDIF
#
# LDAPv3
# base <> with scope baseObject
# filter: (objectclass=*)
# requesting: namingContexts
#

#
dn:
namingContexts: dc=esdebian,dc=org

# search result
search: 2
result: 0 Success

# numResponses: 2
# numEntries: 1
```

Ahora haremos una busqueda en el directorio LDAP autenticado como el usuario admin del LDAP y haremos la buqueda usando como base dc=esdebian,dc=org, esto es para comprobar que la autenticación y nuestras ACLs funcionen correctamente, además de comprobar que el directorio se haya incializado con la estrucutra básica..

```
:~# ldapsearch -x -D "cn=admin,dc=esdebian,dc=org" -b
"dc=esdebian,dc=org" -W
Enter LDAP Password:
# extended LDIF
#
# LDAPv3
# base <dc=esdebian,dc=org> with scope subtree
# filter: (objectclass=*)
# requesting: ALL
#
```

esdebian.org
dn: dc=esdebian,dc=org
objectClass: top
objectClass: dcObject
objectClass: organization
o: esdebian.org
dc: esdebian

admin, esdebian.org
dn: cn=admin,dc=esdebian,dc=org
objectClass: simpleSecurityObject
objectClass: organizationalRole
cn: admin
description: LDAP administrator
userPassword:: e2NyeXB0fWRuZ2QvVWtZMEdzbGc=

search result
search: 2
result: 0 Success

numResponses: 3
numEntries: 2

La herramienta phpldapadmin

Esta es una herramienta alternativa que puede utilizar para gestionar el directorio LDAP. Resulta muy útil pero no es necesaria para realizar la configuración que se plantea en este artículo.

Si desea utilizar esta herramienta basta con instalarla:

aptitude install phpldapadmin

Lo cual instalará una gran cantidad de software como dependencias, entre ellos apache2 y php5.
Una vez instalado el software ya puede acceder desde cualquier navegador mediante:

http://ip_del_servidor/phpldapadmin

Ingresar el nombre de usuario, en este caso cn=admin,dc=esdebian,dc=org, y la clave del mismo.

Configuración de Samba

El archivo /etc/samba/smb.conf

Abrir el archivo y modificar/Agregar los siguientes parametros:

Archivo principal de configuración de Samba

[global]
Juego de caractreres para archivos dos y unix
dos charset = 850
Unix charset = ISO8859-1

Nombre de dominio y nombre netBIOS
workgroup = ESDEBIAN
realm = esdebian.org

Cadena con la cual se identifica al servidor
server string = %h server

Comportamiento frente a usuarios inexistentes
map to guest = Bad User

Archivo de mapeo de nombres de usuarios, este archivo no existe, por lo que debe crearse
El contenido es de la forma usuario = alias
Es util para mapear por ejemplo la cuenta de root con Administrator, la cuenta de
administrador en sistemas windows
username map = /etc/samba/smbusers

Informacion para la utilizacion del directorio LDAP como base de datos de usuarios
passdb backend = ldapsam:ldap://127.0.0.1/
ldap admin dn = cn=admin,dc=esdebian,dc=org
ldap delete dn = Yes
ldap group suffix = ou=group
ldap idmap suffix = ou=idmap
ldap machine suffix = ou=computer
ldap suffix = dc=esdebian,dc=org
ldap ssl = no
ldap user suffix = ou=people
add user script = /usr/sbin/smbldap-useradd -m %u
delete user script = /usr/sbin/smbldap-userdel %u
add group script = /usr/sbin/smbldap-groupadd -p %g
delete group script = /usr/sbin/smbldap-groupdel %g
add user to group script = /usr/sbin/smbldap-groupmod -m %u %g
delete user from group script = /usr/sbin/smbldap-groupmod -x %u %g
set primary group script = /usr/sbin/smbldap-usermod -g %g %u
add machine script = /usr/sbin/smbldap-useradd -w %u

Informacion relacionada con la red (adaptar a sus necesidades)
socket options = TCP_NODELAY SO_RCVBUF=8192 SO_SNDBUF=8192
interfaces = eth0 lo

hosts allow = 127.0.0.1, 192.168.1.0/24
hosts deny = 0.0.0.0
smb ports = 139 445
bind interfaces only = Yes
name resolve order = wins hosts lmhosts bcast
remote announce = 192.168.1.255

Cambio de contraseñas y otras opciones de PDC
pam password change = Yes
passwd program = /usr/sbin/smbldap-passwd -u %u
*passwd chat = *New*password* %n\n *Retype*new*password* %n\n *all*authentication*tokens*updated**

Script de inicio de sesion
logon script = 'logon.bat %U'

Esto hace que los usuarios posean perfiles movibles. Un perfil móvil implica
que gran volumen de información se transmite a través de la red cada vez que
un usuario inicia sesión. Para desactivar los perfiles móviles se debe dejar
el parámetro con un valor vacío, como sigue: logon path = ""
En este ejemplo se permiten perfiles móviles
logon path = \\%N\profiles\%U
logon drive = U:
domain logons = Yes
os level = 65
preferred master = Yes
domain master = Yes
dns proxy = No
wins support = Yes
panic action = /usr/share/samba/panic-action %d
map acl inherit = Yes
case sensitive = No
hide unreadable = Yes

Sincronizar password UNIX con passwords del dominio
unix password sync = Yes

Loggiging
syslog = 0
log file = /var/log/samba/log.%m
max log size = 1000
#
Sincroonizar la hora con el servidor PDC
time server = Yes
#
Mapear atributos de archivo de Unix a Windows

```
# Estas opciones requieren que los parametros create mask y directory mask
# tenga activo el bit de ejecucion para "grupo" y "otros"
map hidden = Yes
map system = Yes

# Recursos compartidos
[homes]
comment = Home Directories
valid users = %S
read only = No
create mask = 0611
directory mask = 0711
browseable = No

[printers]
comment = All Printers
path = /var/spool/samba
create mask = 0611
directory mask = 0711
printable = Yes
browseable = No

[print$]
comment = Printer Drivers
path = /var/lib/samba/printers
create mask = 0611
directory mask = 0711

[netlogon]
path = /var/lib/samba/netlogon
browseable = No
create mask = 0611
directory mask = 0711

[profiles]
path = /var/lib/samba/profiles
force user = %U
read only = No
create mask = 0611
directory mask = 0711
guest ok = Yes
profile acls = Yes
browseable = No
csc policy = disable

[public]
path = /tmp
read only = No
```

guest ok = Yes
create mask = 0611
directory mask = 0711

Se han incluido una serie de explicaciones dentro del archivo a modo de comentario. Seria mas que conveniente que no se limite a copiar el archivo, sino que lea los comentarios.

El archivo /etc/samba/smbusers

Prestar atencion al parametro username map = /etc/samba/smbusers, que indica el archivo de mapeo de nombres de usuarios.

Este archivo no existe por lo que debe crearse, el contenido es de la forma usuario = alias y es util para mapear por ejemplo la cuenta de root con Administrator, la cuenta de administrador en sistemas windows

El contenido de /etc/samba/smbusers podria ser

```
# Archivo de mapeo de usuarios
# Formato: Unix_ID = Windows_ID
#
# Ejemplo:
# root = Administrator
# pepe = "Pepe Parada"
#
root = Administrator
root = Administrador
```

Crear los directorios netlogon y profiles

Crear los directorios especiales netlogon y profile y asignar los permisos adecuados

```
# mkdir -p /var/lib/samba/netlogon /var/lib/samba/profiles
# chown -Rf root:root /var/lib/samba/netlogon /var/lib/samba/profiles
# chmod 1777 /var/lib/samba/profiles
```

Crear scripts de inicio de sesión

Un script de inicio de sesión Windows es un archivo bat con comandos que se ejecutarán del lado del cliente cuando un usuario inicie sesión. El script a utilizarse está definido por el parámetro logon script en la sección [global] de /etc/samba/smb.conf

Se define el paramatro de la siguiente manera

logon script = 'logon.bat %U'

Lo que indica al cliente que debe buscarse el archivo logon.bat en el directorio compartido netlogon.

Además se incluye %U en la definición, %U se reemplazará por el nombre de usuario y éste se pasará por parametro al script. Eso permitirá personalizar la acción dependiendo del usuario que inicia sesión.

Un script de inicio de sesión es utilizado frecuentemente para tareas como:

- Sincronizar la hora con el servidor
- Conectar unidades de red
- Vaciar directorios temporales

Un ejemplo de script de inicio de sesión podría ser:

```
@echo off
net time \\debian-pdc /set /yes

IF %1 == Administrador net use p: \\debian-pdc\root
IF %1 == guest net use p: \\debian-pdc\publico
```

Implementar Politicas de Sistema

A grandes rasgos, Políticas de Sistema es un mecanismo que se emplea en sistemas Microsoft Windows para esablecer ciertas condiciones que se imponen a los usuarios y procesos del equipo.

Cada equipo Windows define su "Políticas de Sistema" local. Cuando un usuario inicia sesión en un equipo que pertenece a un dominio, éste además descarga del servidor, si está disponible, el archivo de políticas Globales y las hace efectivas en el cliente. "Políticas de Sistema" cuenta con una jerarquía, la definición de una política local gobierna a una política global, mientras que una política globál solo se hará efectiva si no está definida esa política localmente.

No es la finalidad de este artículo dar una explicación detallada de Políticas de Sistema, simplemente se dará una breve explicación de como implementarlas en un dominio Samba.

En un entorno de Dominio las Políticas se materializan en un único archivo de nombre NTConfig.POL que se almacenará en el directorio compartido netlogon, de modo que los clientes acceden a este archivo a través de la ruta \\debian-pdc\netlogon\NTConfig.POL

Entonces implementar las políticas es muy simple, la complejidad radica en crear el archivo NTConfig.POL y ajustarlo a nuestras necesidades.

Para esto se debe obtener el software poledit.exe, lo que se hace es, desde un equipo Windows NT compatible:

- Dirigirse al sitio de descargas de Microsoft Windows 2000: http://www.microsoft.com/windows2000/
- Descargar el archivo "Windows 2000 Service Pack 4 Network Install for IT Professionals" (W2kSP4_EN.EXE)
- Descomprimir el paquete con "W2kSP4_EN.EXE /x"
- Ejecutar el archivo "adminpack.msi" para instalar las herramientas
- Correr "poledit.exe"

Ya obtenido el software se puede crear un arhivo de Politicas de manera Standard, se asume que para implementar "Politias de Sistema" usted sabe que son y como se configuran.

Una vez ajustadas las politicas a nuestras necesidades simplemente se debe guardar el archivo como NTConfig.POL y copiarlo al servidor en el directorio compartido netlogon, en nuestro caso /var/lib/samba/netlogon

En http://www.pcc-services.com/custom_poledit.html se explica como configurar políticas de modo mas detallado y además se ofrecen plantillas personalizadas que usted puede utilizar.

NOTA: Esta configuración no es compatible para clientes Windows 98, cuyo archivo de políticas se denomina Config.POL y su estructura es diferente e incompatible. No se tratará en este artículo el caso de clientes Windows 98.

Comprobando la configuración e iniciando el servicio

Para comprobar la configuración de samba basta con ejecutar:

testparm

Si todo es correcto ya se puede reiniciar el servicio:

/etc/init.d/samba restart

Y ahora le se indica a samba la clave del usuario admin especificado en smb.conf para que pueda así acceder y modificar nuestro directorio ldap, en nuestro caso la clave que utilizamos es contraseña

```
# smbpasswd -W
Setting stored password for "cn=admin,dc=esdebian,dc=org" in secrets.tdb
New SMB password: contraseña
Retype new SMB password: contraseña
```

La clave se almacenará en /var/lib/samba/secrets.tdb, asegurémonos que los permisos de dicho archivo sean los adecuados

```
# ls -l /var/lib/samba/secrets.tdb
-rw------- 1 root root 8192 2008-06-18 23:29 /var/lib/samba/secrets.tdb
```

La herramienta SWAT (Samba Web Administration Tool)

Esta herramienta resulta muy útil para configurar un servidor samba, cuenta con una interfaz a la cual se accede a través de un navegador web y permite modificar todos los parametros de smb.conf a la vez que cuenta con toda la información de las paginas man para cada parámetro de configuración.

Al igual que cuando se describió la herramienta phpldapadmin, esta herramienta tampoco es necesaria para la configuración que se propone en este articulo.

Para instalar swat, simplemente hacer:

aptitude install swat

El servicio Swat corre bajo el superserver inetd, por lo que debemos decirle a inetd que active el servicio

update-inetd --enable 'swat'

Ahora ya podemos acceder desde cualquier navegador ingresando en la barra de direcciones http://ip_del_servidor:901

Configuración de smbldap-tools

El archivo de configuración de smbldap-tools es /etc/smbldap-tools/smbldap.conf, en dicho archivo se definen los parámetros básicos como servidor ldap, servidor samba, tipo de comunicación (cifrada o en claro), dominio, SID, etc.

Además necesita contar con el archivo smbldap_bind.conf, en dicho archivo se almacenará en claro la información necesaria para la conexión con el servidor ldap

Obteniendo los archivos de configuración

Si ingresamos al directorio /etc/smbldap-tools/ nos encontraremos con que está vacío, Afortunadamente podemos obtener los archivos de configuración desde los archivos de ejemplo de smbldap-tools

```
# zcat /usr/share/doc/smbldap-tools/examples/smbldap.conf.gz > /etc/smbldap-tools/smbldap.conf
# cp /usr/share/doc/smbldap-tools/examples/smbldap_bind.conf /etc/smbldap-tools/smbldap_bind.conf
```

Los permisos de /etc/smbldap-tools/smbldap.conf y /etc/smbldap-tools/smbldap_bind.conf deberian ser 640, propiedad de root y grupo openldap.

```
# chmod 640 /etc/smbldap-tools/smbldap.conf /etc/smbldap-tools/smbldap_bind.conf
# chown root:openldap /etc/smbldap-tools/smbldap.conf /etc/smbldap-tools/smbldap_bind.conf
```

Obteniendo el SID

El SID (security identifiers) es un identificador único asignado desde su creación a cada objeto dentro de un dominio. Todo objeto en el dominio tiene un SID y, claro, el controlador de dominio también tiene uno.

Para obtener nuestro SID hacer:

net getlocalsid
SID for domain DEBIAN-PDC is: S-1-5-21-3991131808-1853181808-1058153799

Guardar el SID, ya que se necesitara cuando se configura /etc/smbldap-tools/smbldap.conf

NOTA: Probablemente se muestren algunos mensajes de error cuando se ejecuta net getlocalsid, eso se debe a que el paquete smbldap-tools aún no esta configurado. Sin embargo el SID nos será mostrado correctamente

El archivo /etc/smbldap-tools/smbldap.conf

Antes de modificar el archivo original se hace una copia de respaldo mismo

cp /etc/smbldap-tools/smbldap.conf{,.original}

Ahora modificar los siquientes parametros, no olvidar reemplazar el SID por el obtenido con net getlocalsid

```
# El sid obtenido mediante net getlocalsid
SID="S-1-5-21-669132894-2586221759-3914214969"

# Nuestro dominio netBIOS
sambaDomain="ESDEBIAN"

# Informacion del servidor LDAP primario y esclavo
slaveLDAP="127.0.0.1"
slavePort="389"
masterLDAP="127.0.0.1"
masterPort="389"

# No utilizar conexión cifrada
ldapTLS="0"

# Sin importancia ya que no se utiliza TLS
verify="require"
cafile="/etc/smbldap-tools/ca.pem"
clientcert="/etc/smbldap-tools/smbldap-tools.pem"
clientkey="/etc/smbldap-tools/smbldap-tools.key"

# Sufijo LDAP
suffix="dc=esdebian,dc=org"

# Donde se almacenan los usuarios, grupos, computadoras y
idmapdn
usersdn="ou=people,${suffix}"
computersdn="ou=computer,${suffix}"
groupsdn="ou=group,${suffix}"
```

```
idmapdn="ou=idmap,${suffix}"
sambaUnixIdPooldn="sambaDomainName=${sambaDomain},$
{suffix}"

# Default scope
scope="sub"

# Tipo de cifrado UNIX (CRYPT, MD5, SMD5, SSHA, SHA,
CLEARTEXT)
hash_encrypt="CRYPT"
crypt_salt_format="%s"

# Especifico para cuentas UNIX, shell, ruta al home y demás
userLoginShell="/bin/bash"
userHome="/home/%U"
userHomeDirectoryMode="700"
userGecos="System User"
defaultUserGid="513"
defaultComputerGid="515"
skeletonDir="/etc/skel"
defaultMaxPasswordAge="365"

# Configuración especifica para cuentas SAMBA
userSmbHome="\\debian-pdc\%U"
userProfile="\\debian-pdc\profiles\%U"
userHomeDrive="U:"
userScript="logon.bat %U"
mailDomain="esdebian.org"

# Especifico de smbldap-tools
with_smbpasswd="0"
smbpasswd="/usr/bin/smbpasswd"
with_slappasswd="0"
slappasswd="/usr/sbin/slappasswd"
```

El archivo /etc/smbldap-tools/smbldap_bind.conf

/etc/smbldap-tools/smbldap_bind.conf es el archivo de credenciales, en el se almacenará en claro la clave de administrador del servidor ldap. El contenido deberia ser:

```
slaveDN="cn=admin,dc=esdebian,dc=org"
slavePw="ldapadmin"
masterDN="cn=admin,dc=esdebian,dc=org"
masterPw="ldapadmin"
```

Se puede definir un servidor LDAP esclavo, el cual solo se utilizará para consultas, y un servidor primario, el cual se utilizará para escrituras. En este caso, y por el momento, utilizar el mismo servidor.

Está de mas decir que si se utilizan dos servidores LDAP, estos deben estar sincronizados.

Poblando el directorio LDAP con el Schema Samba - La herramienta smbldap-populate

Una vez configuradas las herramientas smbldap-tools, podemos inicializar el dominio

La herramienta smbldap-populate poblará el directorio LDAP con:

- Base DN: dc=esdebian,dc=org
- Unidad Organizativa people(ou=people) para las cuentas Unix/Samba: se crearán por defecto los usuarios root y nobody los cuales serán mapeados al usuarios Administrador y Guest Samba respectivamente.
- Unidad Organizativa Groups (ou=Groups) para los Grupos Unix/Samba: se crearán por defecto los Grupos Predeterminados de un dominio Samba: Domain Admins, Domain Users, Domain Guests, Domain Computers
- Unidad Organizativa Computers (ou=Computers) para las cuentas de Computadoras Windows
- Unidad Organizativa Idmap (ou=Idmap) para los mapeos de Cuentas Unix a Cuentas Samba/Windows (Relaciones UID <-> SID)

Tambien se solicitará que se asigne una contraseña al usuario root del dominio, colocar la clave dominioadmin

Correr el comando smbldap-populate

```
# smbldap-populate
Populating LDAP directory for domain ESDEBIAN (S-1-5-21-669132894-2586221759-3914214969)
(using builtin directory structure)

entry dc=esdebian,dc=org already exist.
adding new entry: ou=people,dc=esdebian,dc=org
adding new entry: ou=group,dc=esdebian,dc=org
adding new entry: ou=computer,dc=esdebian,dc=org
adding new entry: ou=idmap,dc=esdebian,dc=org
adding new entry: uid=root,ou=people,dc=esdebian,dc=org
adding new entry: uid=nobody,ou=people,dc=esdebian,dc=org
adding new entry: cn=Domain Admins,ou=group,dc=esdebian,dc=org
adding new entry: cn=Domain Users,ou=group,dc=esdebian,dc=org
adding new entry: cn=Domain Guests,ou=group,dc=esdebian,dc=org
adding new entry: cn=Domain Computers,ou=group,dc=esdebian,dc=org
adding new entry: cn=Administrators,ou=group,dc=esdebian,dc=org
adding new entry: cn=Account Operators,ou=group,dc=esdebian,dc=org
adding new entry: cn=Print Operators,ou=group,dc=esdebian,dc=org
adding new entry: cn=Backup Operators,ou=group,dc=esdebian,dc=org
adding new entry: cn=Replicators,ou=group,dc=esdebian,dc=org
entry sambaDomainName=ESDEBIAN,dc=esdebian,dc=org already exist. Updating it...
```

Please provide a password for the domain root:
Changing UNIX and samba passwords for root
New password: dominioadmin
Retype new password: dominioadmin

La clave solicitada, dominioadmin, es la clave del usuario Administrador, el administrador del dominio que está mapeado a la cuenta root. Es la clave que se debe utilizar para, por ejemplo, agregar un equipo al dominio.

Testeando la configuración de Samba y ldap

Mediante la herramienta pdbedit testear que el servidor ldap se comporta como se espera, listar los usuarios del directorio

pdbedit -L
root:0:root
nobody:65534:nobody

Ver la información del usuario root

pdbedit -Lv root
Unix username: root
NT username: root
Account Flags: [U]
User SID: S-1-5-21-669132894-2586221759-3914214969-500
Primary Group SID: S-1-5-21-669132894-2586221759-3914214969-513
Full Name: root
Home Directory: \\debian-pdc\root
HomeDir Drive: U:
Logon Script: 'logon.bat root'
Profile Path: \\debian-pdc\profiles\root
Domain: ESDEBIAN
Account desc:
Workstations:
Munged dial:
Logon time: 0
Logoff time: never
Kickoff time: never
Password last set: lun, 18 ene 2010 09:07:47 ART
Password can change: lun, 18 ene 2010 09:07:47 ART
Password must change: never
Last bad password : 0
Bad password count : 0
Logon hours : FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

La cuenta root que se creo en el directorio LDAP, es la que se usará como usuario root local y como administrador de dominio, asignarle un home y un shell:

```
# smbldap-usermod -d /root -s /bin/bash root
```

Probar ahora conectarnos mediante smbclient:

```
# smbclient //localhost/netlogon -U root
Enter root's password: dominioadmin
Domain=[ESDEBIAN] OS=[Unix] Server=[Samba 3.2.5]
smb: \>
```

Podemos testear el funcionamiento de los alias de root que se definió antes en el archivo /etc/samba/smbusers, Administrador y Administrator

```
~# smbclient //localhost/netlogon -U administrador
Enter administrador's password: dominioadmin
Domain=[ESDEBIAN] OS=[Unix] Server=[Samba 3.2.5]
smb: \> quit
~# smbclient //localhost/netlogon -U administrator
Enter administrator's password: dominioadmin
Domain=[ESDEBIAN] OS=[Unix] Server=[Samba 3.2.5]
smb: \> quit
~#
```

Configuración de PAM/NSS

El servicio NSS (Name Service Switch) provee una interface para configurar y acceder a diferentes bases de datos de cuentas de usuarios, la forma mas básica y conocida es acceder a la información de los usuarios locales mediante los archivos /etc/passwd y /etc/shadow, /etc/group, /etc/hosts, etc. Sin embargo se pueden implementar otros mecanismos para ello.

PAM (Pluggable Authentication Modules) es un mecanismo de autenticación flexible que permite abstraer las aplicaciones y otro software del proceso de identificación. El módulo PAM, según como se haya configurado, accederá a la información y determinará su identidad, autenticidad, privilegios y limitaciones entre otras cosas.

Entonces, la combinación PAM/NSS nos provee una capa de abstracción que nos permite obtener la información de usuarios y su identidad sin importar que dicha información sea almacenada en un simple archivo de texto plano (/etc/passwd) o un complejo directorio ldap.

Tomar en cuenta que, al acceder tanto PAM como NSS a la misma base de datos, la configuración de ambos es la misma.

Configuración de libnss-ldap

El paquete libnss-ldap es el nexo que permitirá el servicio NSS acceder y utilizar la información en el directorio LDAP. Reconfiguremos el paquete mediante:

dpkg-reconfigure libnss-ldap

Y responder a las siguientes preguntas:

- Identificador de recursos para el servidor LDAP: ldap://127.0.0.1/
- El nombre distintivo (DN) de la base de búsquedas: dc=esdebian,dc=org
- Versión de LDAP a utilizar: 3
- ¿Hace falta un usuario para acceder a la base de datos LDAP?: No
- ¿Dar privilegios especiales de LDAP para root?: Sí
- ¿Desea hacer que la configuración sólo la pueda leer o escribir el propietario? Sí
- Cuenta LDAP para root: cn=admin,dc=esdebian,dc=org
- Contraseña para la cuenta LDAP de root: contraseña

El demonio nscd (Name Service Cache Daemon) es una caché de nombres para el servicio NSS. Acelera de forma significativa las consultas pero puede hacernos pasar malos ratos durante la configuración. Detengamos el servicio por ahora

/etc/init.d/nscd stop

El archivo /etc/libnss-ldap.conf

/etc/libnss-ldap.conf es el archivo de configuración de libnss-ldap. Allí se podrían almacenar en claro alguna clave, no en nuestro caso pero debe igualmente asegurarse que los permisos son restrictivos. Propietario root, grupo root y permisos 600.

Buscar y reemplazar el contenido de las siguientes lineas:

```
base dc=esdebian,dc=org
uri ldap://127.0.0.1/
ldap_version 3
rootbinddn cn=admin,dc=esdebian,dc=org
bind_policy soft
pam_password crypt
nss_base_passwd dc=esdebian,dc=org?sub
nss_base_shadow dc=esdebian,dc=org?sub
nss_base_group ou=group,dc=esdebian,dc=org?one
```

El archivo /etc/libnss-ldap.secret

/etc/libnss-ldap.secret es el archivo de credenciales. Allí se almacena en claro la clave de root del servidor ldap, asegurar de que los permisos son restrictivos .

Propietario root, grupo root y permisos 600 . Verificar el contenido del archivo:

cat /etc/libnss-ldap.secret
contraseña

Configurando el paquete libpam-ldap

Reconfiguremos el paquete libpam-ldap

dpkg-reconfigure libpam-ldap

La configuración es similar a libnss-ldap, respondamos a las siguientes preguntas:

- Identificador de recursos para el servidor LDAP: ldap://127.0.0.1/
- El nombre distintivo (DN) de la base de búsquedas: dc=esdebian,dc=org
- Versión de LDAP a utilizar: 3
- Crear un administrador de la base de datos local: Sí
- ¿Hace falta un usuario para acceder a la base de datos LDAP?: No
- Cuenta LDAP para root: cn=admin,dc=esdebian,dc=org
- Contraseña para la cuenta LDAP de root: contraseña
- Cifrado local a utilizar cuando se cambian las contraseñas: crypt

El archivo /etc/pam_ldap.conf

/etc/pam_ldap.conf es el archivo de configuración de libpam-ldap. Buscar y reemplazar el contenido de las siguientes lineas:

base dc=esdebian,dc=org
uri ldap://127.0.0.1/
ldap_version 3
rootbinddn cn=admin,dc=esdebian,dc=org
bind_policy soft
pam_password crypt
nss_base_passwd dc=esdebian,dc=org?sub
nss_base_shadow dc=esdebian,dc=org?sub
nss_base_group ou=group,dc=esdebian,dc=org?one

El archivo /etc/pam_ldap.secret

/etc/pam_ldap.secret es el archivo de credenciales. Allí se almacena en claro la clave de root del servidor ldap, asegurar de que los permisos son restrictivos .

Propietario root, grupo root y permisos 600 . Verificar el contenido del archivo:

cat /etc/pam_ldap.secret
ldapadmin

El archivo /etc/nsswitch.conf

/etc/nsswitch.conf es el archivo de configuración del servicio NSS, él determina cuales son los orígenes que se utilizarán para obtener la información. Teniendo ya configurado libnss-ldap solo modificar las siguientes lineas agregando el origen ldap:

passwd: files ldap
group: files ldap
shadow: files ldap

Esto le indica al servicio NSS que debe utilizar el origen files (/etc/passwd, /etc/group y /etc/shadow respectivamente) y el origen ldap (mediante el modulo libnss-ldap)

Ya teniendo samba configurado como servidor de nombres wins podemos agregar también el soporte para resolucion de nombres de hosts mediante ese servicio

hosts: files wins dns

Aqui el orden es mas importante, esto indica que deben resolverse nombre de host mediante files (/etc/hosts), luego se utilizará wins (samba) y finalmente mediante el DNS resolver

Con esta configuración ya debe estar en condiciones de obtener información de los usuarios desde todos los orígenes configurados, no está de mas recordar que se debe detener o al menos reiniciar el servicio nscd.

Comprobar que el usuario root pertenece al grupo Domain Admins:

```
# id root
uid=0(root) gid=0(root) grupos=0(root),512(Domain Admins)
```

Comprobar las dos entradas para el usuario root, la de /etc/passwd y la de ldap

```
# getent passwd | grep root
root:x:0:0:root:/root:/bin/bash
root:x:0:0:Netbios Domain Administrator:/home/root:/bin/bash
```

Comprobar los grupos del dominio

```
# getent group | grep -E 'root|Domain'
root:x:0:
Domain Admins:*:512:root
Domain Users:*:513:
Domain Guests:*:514:
Domain Computers:*:515:
```

Una cosa interesante que pasa cuando el servidor tiene más servicios que prestar como bases de datos, web, etc; es que cuando se reinicia la PC sale un error como el siguiente:

nss_ldap: failed to bind to LDAP server ldap://127.0.0.1/: Can't contact LDAP server

Ese es uno de los muchos errores que aparecerán, pero no afecta en nada el funcionamiento del PDC, aunque para solucionar esto y todo se vea bien durante el reinicio del servidor se deben agregar los siguietes grupos y usarios en la consola:

addgroup --system nvram
addgroup --system rdma
addgroup --system fuse
addgroup --system kvm
addgroup --system scanner
adduser --system --group --shell /usr/sbin/nologin --home /var/lib/tpm tss

Configurando PAM

Ahora configuremos el módulo PAM para poder acceder localmente con usuarios del Dominio Antes de modificar nada, seria buena idea respaldar la configuración de pam

cp -a /etc/pam.d /etc/pam.d.ORIGINAL

El archivo /etc/pam.d/common-auth

Lo que denominarías el bloque primario, si cualquiera de los dos módulos tiene
exito se salta la ejecucion del modulo pam_deny.so
auth [success=2 default=ignore] pam_unix.so nullok_secure
auth [success=1 default=ignore] pam_ldap.so use_first_pass

Si no se salta este punto la autenticacion falla siempre
auth requisite pam_deny.so

aún cuando la autenticacion tubiera éxito los modulos del bloque
primario podrian no retornar un código positivo, esto soluciona eso
asegurando un valor positivo si no lo era antes
auth required pam_permit.so

El archivo /etc/pam.d/common-account

Lo que denominarias el bloque primario, si cualquiera de los dos módulos tiene
exito se salta la ejecucion del modulo pam_deny.so
account [success=2 new_authtok_reqd=done default=ignore] pam_unix.so
account [success=1 default=ignore] pam_ldap.so

Si no se salta este punto la autenticacion falla siempre
account requisite pam_deny.so

aún cuando la autenticacion tubiera éxito los modulos del bloque

primario podrian no retornar un código positivo, esto soluciona eso
asegurando un valor positivo si no lo era antes
account required pam_permit.so

El archivo /etc/pam.d/common-password

Es buena idea que se tenga restriciones sobre los passwords
cracklib permite controlar longitudes minimas y fortaleza de la clave
ante ataques de diccionario
Solo no debe ovidarse instalar cracklib, simplemente aptitude install libpam-cracklib
password required pam_cracklib.so difok=2 minlen=8 dcredit=2 ocredit=2 retry=3

Lo que denominarias el bloque primario, si cualquiera de los dos módulos tiene
exito se salta la ejecucion del modulo pam_deny.so
password [success=2 default=ignore] pam_unix.so obscure crypt
password [success=1 user_unknown=ignore default=die] pam_ldap.so use_authtok try_first_pass

Si no se salta este punto la autenticacion falla siempre
password requisite pam_deny.so

aún cuando la autenticacion tubiera éxito los modulos del bloque
primario podrian no retornar un código positivo, esto soluciona eso
asegurando un valor positivo si no lo era antes
password required pam_permit.so

El archivo /etc/pam.d/common-session

Lo que denominarias el bloque primario, si el usuario está correctamente autenticado
se salta la ejecucion del modulo pam_deny.so (recordar que este es un apartado de
sesion, por lo tanto lo que se controla no es el acceso sino los privilegios efectivos
de un usuario ya autenticado)
session [default=1] pam_permit.so

Si no se salta este punto la autenticacion falla siempre
session requisite pam_deny.so

aún cuando la autenticacion tubiera éxito los modulos del bloque
primario podrian no retornar un código positivo, esto soluciona eso
asegurando un valor positivo si no lo era antes
session required pam_permit.so

Se chequean los privilegios y restricciones
session required pam_unix.so
session optional pam_ldap.so

Creación de usuarios y grupos en el directorio LDAP mediante smbldap-tools

Llegado a este punto ya se debería estar en condiciones de ingresar al servidor mediante usuarios del Dominio, los cuales autenticarán mediante PAM, accediendo éste a la información almacenada en el directorio LDAP.

Pero para probar si es posible ingresar al sistema local con un usuario del dominio primero se debe crear uno

Crear usuarios del dominio

Para esto nos valdremos de las herramientas provistas por smbldap-tools.
Mediante la herramienta smbldap-useradd agregamos al usuario pepe al dominio

smbldap-useradd -a -m -P pepe
Cannot confirm uidNumber 1000 is free: checking for the next one
Changing UNIX and samba passwords for pepe
New password: 123456
Retype new password: 123456

- La opción -a indica que es un usuario Windows
- La opción -m indica que debe crearse el home del usuario e inicializarlo con el contenido de /etc/skel que es el comportamiento standard para la creación de una cuenta UNIX. Esta opción no siempre es necesario incluirla ya puede pasar que no todos los usuarios del dominio necesiten un HOME en el servidor
- La opción -P indica que se solicitará y establecerá un password al usuario

man smbldap-useradd para mas información acerca de las opciones del comando

La contraseña de pepe es 123456, es una contraseña muy simple que el sistema nos ha dejado crear porque somos root, si pepe desea en el futuro cambiar su contraseña deberá cumplir las especificaciones del archivo /etc/pam.d/common-password impuestas por el módulo pam_cracklib

También vemos el mensaje "Cannot confirm uidNumber 1000 is free: checking for the next one". Ese mensaje solo aparecerá ésta primera vez, de ahora en más el último UID se almacenará en sambaUnixIdPooldn como lo establece el archivo /etc/smbldap-tools/smbldap.conf y el siguiente UID libre se determinará en base a eso. Cuando se agrege el primer grupo sucederá lo mismo.

Se puede verificar que el usuario se ha creado pero no existe localmente:

:~# cat /etc/passwd | grep pepe
:~# getent passwd | grep pepe
pepe:x:1003:513:System User:/home/pepe:/bin/bash
:~#

El paquete libpam-dotfile nos provee la herramienta pamtest que nos permite verificar el funcionamiento de los mecanismos de autenticación. Probar con el usuario pepe

```
:~# pamtest passwd pepe
Trying to authenticate <pepe> for service <passwd>.
Password: 123456
Authentication successful.
```

Los mecanismos de autenticación funcionan correctamente, accedamos al sistema con el usuario pepe:

```
debian-pdc login: pepe
Password: 123456
Linux debian-pdc 2.6.26-2-686 #1 SMP Wed Nov 4 20:45:37 UTC 2009 i686

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
I have no name!@debian-pdc:~$
```

Y se logra acceder al equipo mediante una cuenta ldap. Llama la atencion ese "I have no name!" en el prompt, si se hace

```
$ whoami
whoami: cannot find name for user ID 1001
```

Por algún motivo y aún con el demonio nscd detenido pasa esto, pero si se reinicia el demonio

```
# /etc/init.d/nscd restart
```

ahora si

```
$ whoami
pepe
```

Crear un usuario administrador del dominio

A diferencia de los sistemas Linux/Unix, donde el administrador del dominio es root y su UID es 0, en un Dominio Windows para ser administrador del dominio basta con pertenecer al grupo Domain Admins, cuyo GID es 512.

Por lo tanto, para crear un usuario administrador del dominio, basta con indicarle a smbldap-adduser que lo incluya en el grupo 512

```
# smbldap-useradd -a -m -G 512 -P adminnuevo
Changing UNIX and samba passwords for adminnuevo
```

New password: adminnuevo
Retype new password: adminnuevo

También se puede ingresar al sistema mediante éste usuario, y se podra comprobar que es solo un administrador del dominio pero no tiene permisos de root (su prompt no es #)

$ id adminnuevo
uid=1002(adminnuevo) gid=513(Domain Users) grupos=512(Domain Admins),513(Domain Users)

Crear un Grupo en el dominio

Simplemente haciendo:

:~# smbldap-groupadd -a "Grupo Nuevo"
Cannot confirm gidNumber 1000 is free: checking for the next one
Cannot confirm gidNumber 1001 is free: checking for the next one
Cannot confirm gidNumber 1002 is free: checking for the next one
:~#

Podemos verificar que el grupo existe, pero no localmente

:~# cat /etc/group | grep "Grupo Nuevo"
:~# getent group | grep "Grupo Nuevo"
Grupo Nuevo::1003:*
:~#

Eliminar usuarios y grupos del dominio

Probemos eliminar el grupo recién creado:

:~# smbldap-groupdel "Grupo Nuevo"
:~#

Y el grupo ya no existe

:~# getent group | grep "Grupo Nuevo"
:~#

Eliminemos el usuario pepe que creamos anteriormente

:~# smbldap-userdel -r pepe
:~#

La opción -r indica que debe eliminarse el home del usuario
Se Puede verificar que el usuario ya no existe

:~# getent passwd | grep pepe
:~#

Agregar los usuarios y grupos Linux/Unix locales al directorio LDAP

Para la migracion de usuarios y grupos Unix al ldap se utilizarán herramientas provistas por el paquete smbldap-tools

Dichas herramientas no son provistas como parte del paquete de software sino en la parte de documentación. Las herramientas se encuentran en los archivos:
/usr/share/doc/smbldap-tools/examples/migration_scripts/smbldap-migrate-unix-accounts.gz
/usr/share/doc/smbldap-tools/examples/migration_scripts/smbldap-migrate-unix-groups.gz

Debemos descomprimir estos archivos y hacerlos ejecutables. Los copiaremos a alguna ubicación que figure en nuestro path como ser /usr/sbin

zcat /usr/share/doc/smbldap-tools/examples/migration_scripts/smbldap-migrate-unix-accounts.gz > /usr/sbin/smbldap-migrate-unix-accounts
zcat /usr/share/doc/smbldap-tools/examples/migration_scripts/smbldap-migrate-unix-groups.gz > /usr/sbin/smbldap-migrate-unix-groups
chmod 755 /usr/sbin/smbldap-migrate-unix-accounts /usr/sbin/smbldap-migrate-unix-groups

Migrando usuarios locales al directorio LDAP

El script smbldap-migrate-unix-accounts está preparado para recibir como parametro el archivo /etc/passwd y /etc/shadow, sin embargo no se desea exportar todos los usuarios.

Usuarios como backup, bin o daemon no nos son necesarios en el directorio ldap, tampoco se puede migrar root y nobody porque esos usuarios ya existen en el directorio.

Solo migrar los usuarios necesarios, copiar /etc/passwd y /etc/shadow a una ubicaion temporal:

cp /etc/passwd /etc/shadow /tmp/

Quitar de esos archivos temporales los usuarios que no se desea migrar y entonces si correr la herramienta de migración:

smbldap-migrate-unix-accounts -a -P /tmp/passwd -S /tmp/shadow

Migrando grupos locales al directorio LDAP

Nuevamente copiar /etc/group a una ubicacion temporal:

cp /etc/group /tmp/

Editar el archivo dejando los grupos que se desea migrar (podria no migrar los grupos root, bin y daemon) y entonces correr la herramienta de migración:

smbldap-migrate-unix-groups -a -G /tmp/group

Seguramente se querrá que los usuarios Samba pertenezcan a ciertos grupos Unix, por ejemplo los grupos audio, video, cdrom, plugdev, floppy, etc. para tener esos privilegios cuando accedan desde terminales Linux/Unix. Para ese hay que agregarlos a dichos grupos, por ejemplo al usuario adminnuevo:

smbldap-usermod --group audio,video,floppy,cdrom,plugdev,"Domain Admins","Domain Users" adminnuevo

y se puede ver que adminnuevo ya es miembro de esos grupos:

id adminnuevo
uid=1002(adminnuevo) gid=513(Domain Users) grupos=513(Domain Users),512(Domain Admins),24(cdrom),25(floppy),29(audio),44(video),46(plugdev)

NOTA: Recordar detener el demonio nscd para hacer estas pruebas

Agregar equipos Windows al Dominio

Ya realizadas las configuraciones, el servidor se comporta ahora como un controlador de Dominio NT, ya debería ser posible unir terminales Windows al Dominio.

Requisitos del sistema Windows

El sistema a agregar al dominio será un Windows XP Professional.

Antes que nada se debe estar consiente que el PDC no es un Dominio Active Directory, por lo que:

- No se utiliza el mecanismo de autenticación MIT Kerberos: samba puede ser cliente en un dominio Active Directory y utilizar Kerberos para la autenticación, pero no puede utilizar este mecanismo cuando es un PDC.
- No incorpora un servidor DNS: a diferencia de un dominio NT, donde los nombres se resuelven mediante wins, un dominio Active Directory es gobernado por una implementación DNS. Si bien Active Directory soporta wins (no por defecto, debe instalarse) se basa en DNS para ubicar objetos en la red y esto requiere que los clientes tengan al controlador de dominio como servidor DNS primario.

El no utilizar Kerberos no es un problema, los clientes se autenticarán mediante NTLM, un protocolo lo suficientemente seguro.

El no ser un servidor DNS resulta para nosotros en una mínima configuración más, al no contar con un servidor DNS el dominio esdebian.org no se resuelve con la IP del servidor.

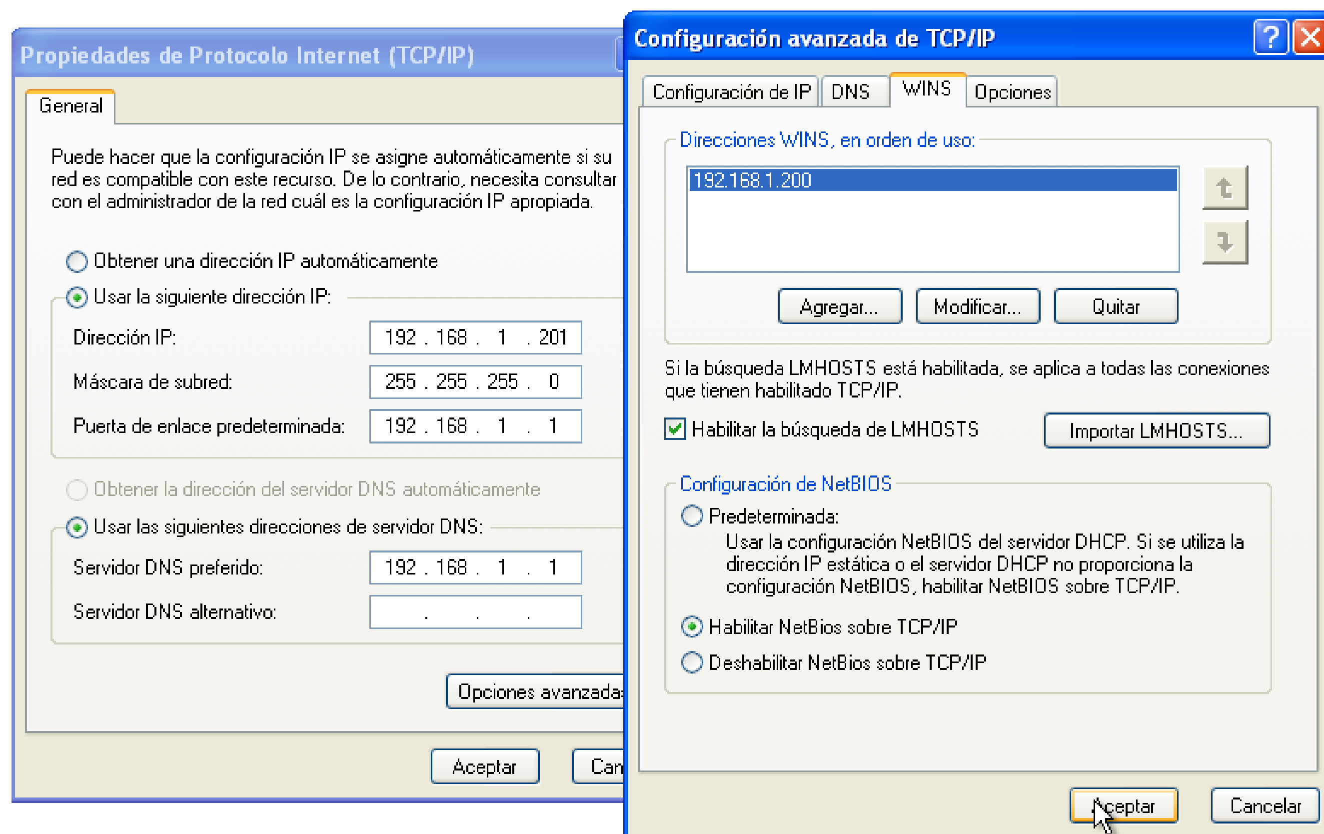
Esto no supone un problema, simplemente se debe utilizar nombres NetBIOS en lugar de nombres DNS, por lo tanto los clientes no podrán agregarse al dominio esdebian.org sino que deberán hacerlo al dominio NetBIOS ESDEBIAN.

Se debe configurar a los clientes para que utilicen el Samba como servidor de nombres WINS.

NOTA: hacer que el servidor sea también un servidor DNS no es nada complicado, pero esa configuración no se tratará en este articulo.

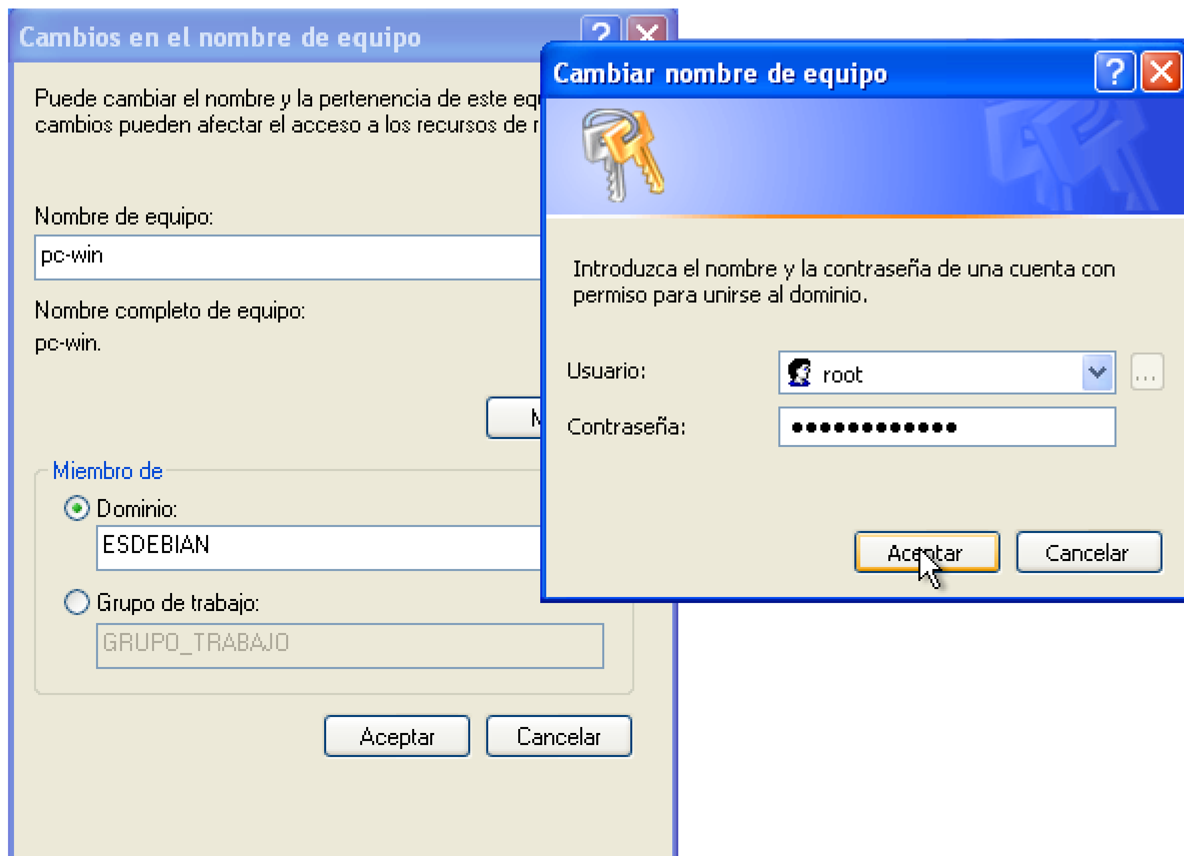
Configuración de red

Asignar al cliente una dirección IP dentro del segmento de red del servidor y configurar el servidor wins para que sea el servidor Samba. Debemos habilitar también NetBIOS sobre TCP/IP:

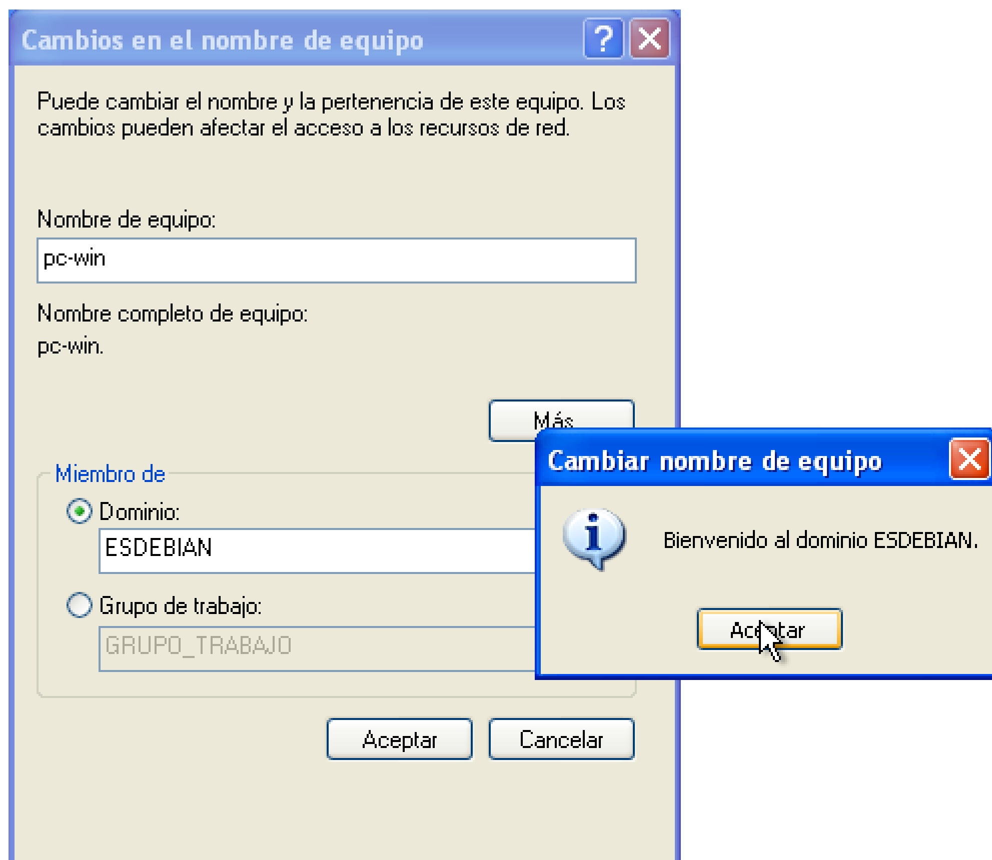


Unir el equipo con Windows XP Professional al dominio Samba

Agregar el equipo al dominio de manera standard. Se utiliza la cuenta de root, cuyo password era dominioadmin



Y se debe recibir el mensaje de bienvenida al Dominio ESDEBIAN



Eliminar el equipo del dominio

Un equipo dentro del dominio es un usuario dentro de la ou=Computers, el nombre de usuario es el nombre del equipo con la diferencia que su nombre finaliza con un $

Se puede ver con

getent passwd | grep Computer
pc-win$::1004:515:Computer:/dev/null:/bin/false*

Mas información con

:~# pdbedit -Lv pc-win$
Unix username: pc-win$
NT username: pc-win$
Account Flags: [W   ]
User SID: S-1-5-21-669132894-2586221759-3914214969-1001
Primary Group SID: S-1-5-21-669132894-2586221759-3914214969-515
Full Name: PC-WIN$
*Home Directory: \\debian-pdc\pc-win_*

HomeDir Drive: U:
Logon Script: 'logon.bat pc-win_'
*Profile Path: \\debian-pdc\profiles\pc-win_*
Domain: ESDEBIAN
Account desc: Computer
Workstations:
Munged dial:
Logon time: 0
Logoff time: never
Kickoff time: never
Password last set: mar, 19 ene 2010 07:30:23 ART
Password can change: mar, 19 ene 2010 07:30:23 ART
Password must change: never
Last bad password : 0
Bad password count : 0
Logon hours : FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

Por lo tanto eliminar un equipo del dominio es lo mismo que eliminar un usuario, se lo hace mediante la herramienta smbldap-userdel

:~# smbldap-userdel pc-win$

Notas para equipos con Windows 7

Para unir equipos con windows 7 al dominio hay que hacer unos cambios en el registro de los clientes, así como actualizar samba a la versión 3.4 de Debian-backports. Para ello añadir la siguiente linea al archivo sources.list:

deb http://backports.debian.org/debian-backports lenny-backports main

Hacer un update e instalar samba:

aptitude -t lenny-backports install samba

Instalará los paquetes actualizados y desinstalará smbfs (no problem).

A continuación se debe crear dos claves de registro en las máquinas con Windows 7 que se unirán al dominio:

HKLM\System\CCS\Services\LanmanWorkstation\Parameters
DWORD DomainCompatibilityMode = 1
DWORD DNSNameResolutionRequired = 0

Ahora ya se puede unir las máquinas al dominio y abrir sesión en el mismo.

Configurando PAM/NSS de clientes Linux/Unix para que utilicen la Autenticación centralizada.

Una vez que se hace login en el equipo cliente Linux/Unix se esta listo para configurar PAM/NSS para hacer que dicho equipo utilice el servidor PDC para la autenticación y manejo de permisos.

Para el ejemplo utilizar Debian/GNU Linux 5.0 como Sistema Operativo para el equipo cliente pero cualquier cliente Linux/Unix debe porder ser configurado. La única diferencia destacable en la configuracion es que en sistemas no Debian el archivo /etc/libnss-ldap.conf suele llamarse simplemente /etc/ldap.conf

Instalar en el cliente el software necesario:

aptitude install libnss-ldap libpam-ldap libpam-cracklib

y responder a las siguientes preguntas:

Configuración de libnss-ldap:

- Identificador de recursos para el servidor LDAP: ldap://192.168.1.200/
- El nombre distintivo (DN) de la base de búsquedas: dc=esdebian,dc=org
- Versión de LDAP a utilizar: 3
- Cuenta LDAP para root: cn=admin,dc=esdebian,dc=org
- Contraseña para la cuenta LDAP de root: dejar vacío

Configuración de libpam-ldap:

- Crear un administrador de la base de datos local: No
- ¿Hace falta un usuario para acceder a la base de datos LDAP? : No

No olvidar la seguridad

Es importante destacar que cuando se preguntó "Contraseña para la cuenta LDAP de root:" durante la configuración de libnss-ldap no se respondió nada.

La contraseña de root para la cuenta ldap es la que definimos en el servidor, en este caso era contraseña, y la utiliza entre otras cosas samba para modificar la base de datos agregando usuarios y demás. Sin embargo el servidor ldap es accesible desde toda la red y una red con cientos de clientes Linux/Unix que tengan almacenada la contraseña de root para acceder al servidor ldap no parece muy inteligente.

Cualquier equipo que se vea comprometido tendrá la posibilidad de acceder y modificar sin restricciones la base de datos.

El parámetro "Cuenta LDAP para root:" también lo hemos establecido pero no tiene sentido alguno, cuando se revise los archivos de configuración comentar la linea.

Los archivos de configuración de PAM y NSS

PAM y NSS utilizarán la misma base de datos ldap, por la tanto su configuración es igual.

Editar los archivos /etc/libnss-ldap.conf y /etc/pam_ldap.conf, y reemplazar con lo siguiente (la configuración es muy similar a la del servidor):

```
# DN base
base dc=esdebian,dc=org

# URI del servidor ldap, en nuestro caso es 192.168.1.200
uri ldap://192.168.1.200/

# Version de ldap a utilizar
ldap_version 3

# Cuenta de root ldap
# Esta linea no es necesaria, la comentamos o borramos
# rootbinddn cn=admin,dc=esdebian,dc=org

bind_policy soft
pam_password crypt

nss_base_passwd dc=esdebian,dc=org?sub
nss_base_shadow dc=esdebian,dc=org?sub
nss_base_group ou=group,dc=esdebian,dc=org?one
```

Modificar /etc/nsswitch.conf agregando ldap para las búsquedas y wins para resolver nombres:

```
passwd:     files ldap
group:      files ldap
shadow:     files ldap
hosts:      files wins dns
```

Hacer un backup de la configuración de PAM

```
# cp -a /etc/pam.d{,.ORIGINAL}
```

Modificar los archivos de configuración de PAM:

/etc/pam.d/common-auth

```
auth   [success=2 default=ignore]   pam_unix.so nullok_secure
auth   [success=1 default=ignore]   pam_ldap.so use_first_pass
auth   requisite                    pam_deny.so
auth   required                     pam_permit.so
```

/etc/pam.d/common-account

account [success=2 new_authtok_reqd=done default=ignore] pam_unix.so
account [success=1 default=ignore] pam_ldap.so
account requisite pam_deny.so
account required pam_permit.so

/etc/pam.d/common-password

password required pam_cracklib.so difok=2 minlen=8 dcredit=2 ocredit=2 retry=3
password [success=2 default=ignore] pam_unix.so obscure crypt
password [success=1 user_unknown=ignore default=die] pam_ldap.so use_authtok try_first_pass
password requisite pam_deny.so
password required pam_permit.so

/etc/pam.d/common-session

La configuración de este archivo difiere de la del servidor PDC. En el servidor se puede crear las cuentas de usuario mediante smbldap-useradd agregando la opción -m de modo que el home de cada usuario se crea junto con el usuario.

En los clientes el home de cada usuario no existe, por lo que se utiliza el módulo pam_mkhomedir.so para crear el home del usuario en caso que no exista. Eso es lo que motiva agregar la primer linea del archivo

session required pam_mkhomedir.so
session [default=1] pam_permit.so
session requisite pam_deny.so
session required pam_permit.so
session required pam_unix.so
session optional pam_ldap.so

Probando la configuración

Primero que nada, detener el demonio nscd

/etc/init.d/nscd stop

Ahora ya se puede hacer login en los clientes Linux/Unix con algún usuario del dominio Samba, utilizar la cuenta de adminnuevo que se creo anteriormente:

debian-cli login: adminnuevo
Password:

Linux debian-cli 2.6.26-2-686 #1 SMP Wed Nov 4 20:45:37 UTC 2009

i686

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc//copyright.*

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Creating directory '/home/adminnuevo'
adminnuevo@debian-cli:~$

Y se ha logrado hacer login con un usuario del dominio

Ver también como el módulo pam_mkhomedir.so definido en /etc/pam.d/common-session ha creado el home del usuario en el equipo local. Esto solo sucede para el primer login

Referencias:

1. Presentación Curso ldap-samba del Centro de Estudios Tecnológicos Avanzados (CENTEC).
2. http://www.esdebian.org/wiki/controlador-primario-dominio-pdc-debian-lenny-50-mediante-samba-pamnss-openldap#11
3. http://amd.co.at/adminwiki/PAM/NSS